

Sentinel

<Delve into Web Dev #2>

DEFENDING OUR DIGITAL WAY OF LIFE

Learning Objective

Understand **nodes** and **children**

Use different **methods** to select children

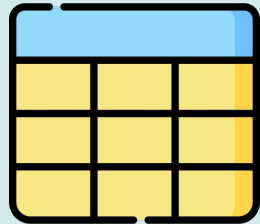
Assign **class** to elements

Use different **selector methods** to get elements

What we want to do

To develop a frequency analysis tool we need to:

Populate a table with the analysis result



Parse the HTML table and use it as the key



Both of these are going to require some deeper understanding of the DOM and how to manipulate it!

Recap: DOM

DOM: Document Object Model

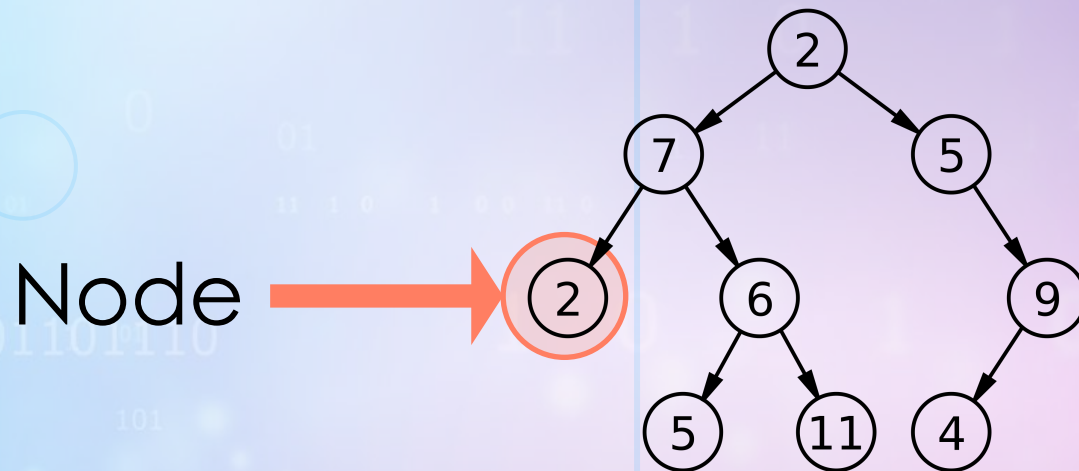


Fancy way of saying “the hierarchy of all the elements of our HTML document”

DOM - Continued

First, let's talk about nodes!

Node: Data points that are linked together to form a structure. Most nodes are arranged into trees.

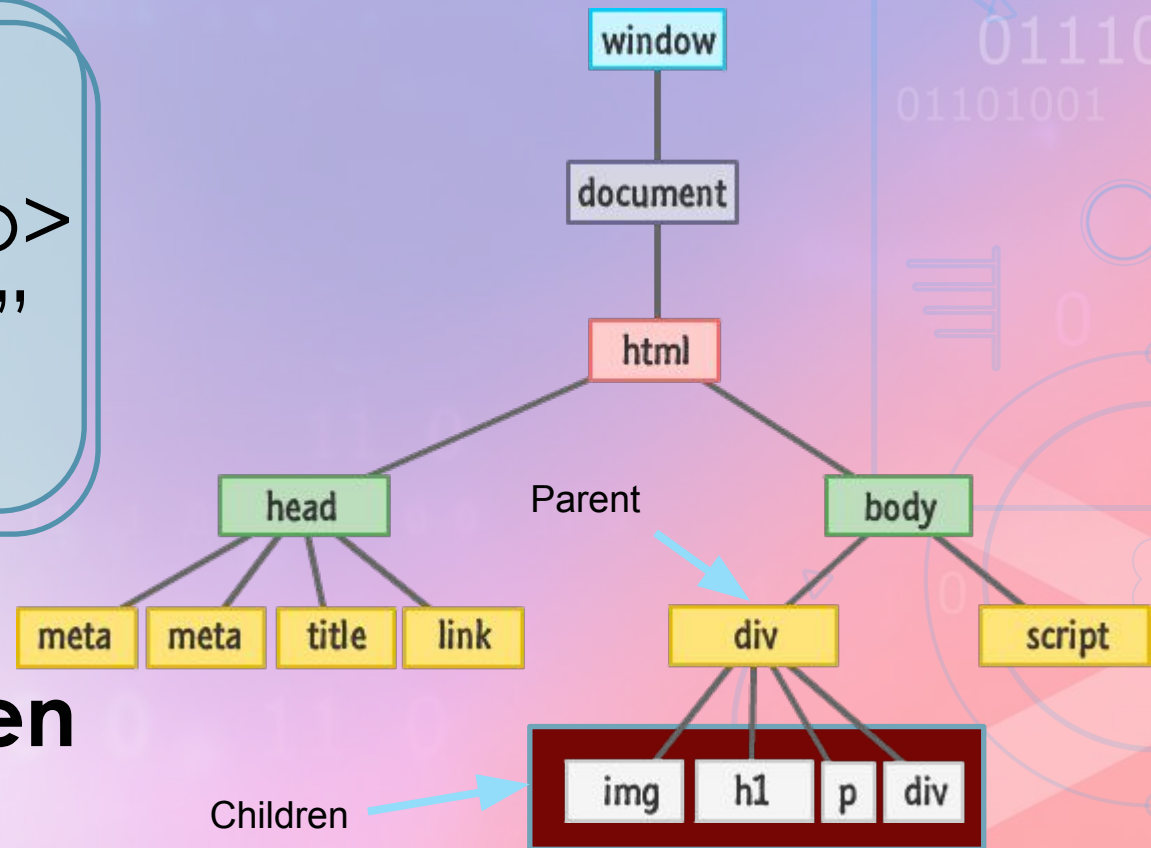


We will be talking about nodes in the **DOM**!

DOM - Continued

Every node is one of:

- Element – like `<p></p>`
- Attribute – like `href=""`
- Text – like `"bla"`



Nodes can have **children**

HTMLCollection

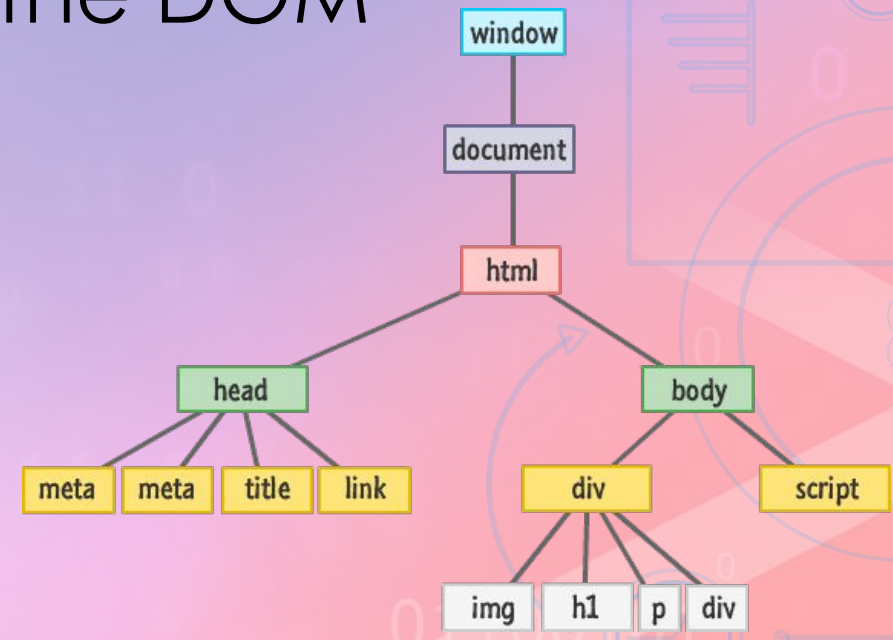
Like an Array of elements in document order

We can use them to traverse the DOM

Let's investigate:



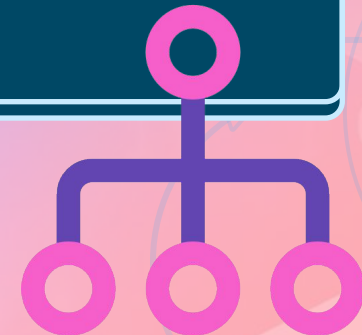
```
document.body.children
```



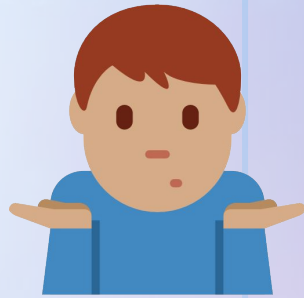
Iterating Elements

Same as Arrays!

```
for (const element of document.body.children) {  
  console.log(element.tagName)  
}
```



Reminder: Getting Elements

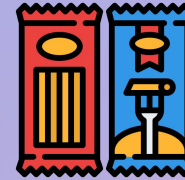
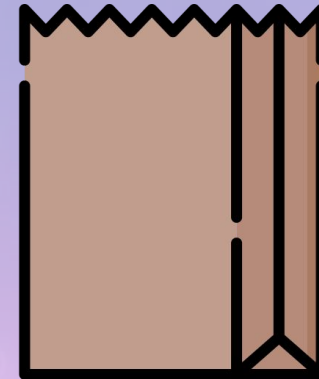


How do we select a specific element?

```
document.getElementById("element-id")
```

Shopping List

```
<ul id="shopping-list">  
  <li>Pasta</li>  
  <li>Tomatoes</li>  
  <li>Basil</li>  
  <li>Olive Oil</li>  
</ul>
```



```
let listEl = document.getElementById("shopping-list")
```

Shopping List

We can print each item of the list:

```
let listEl = document.getElementById("shopping-list")  
  
for (const listItem of listEl.children) {  
  console.log(listItem.innerText)  
}
```

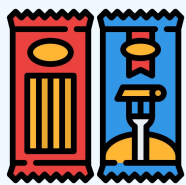


Shopping List

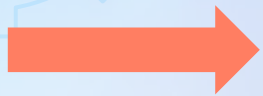
Or save it in an Array for manipulation

```
let listEl = document.getElementById("shopping-list")
let shoppingList = []

for (const listItem of listEl.children) {
  shoppingList.push(listItem.innerText)
}
```



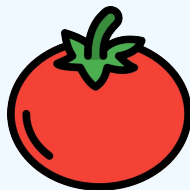
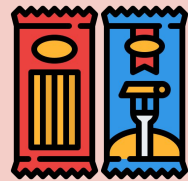
First and Last Children



```
listEl.firstChild  
// <li>Pasta</li>
```

```
listEl.lastElementChild  
// <li>Olive Oil</li>
```

```
listEl.children[2]  
// <li>Basil</li>
```



Selecting By Tag Name

Let's print out all of the image URLs in a webpage

```
let imageElements = document.getElementsByTagName("img")

for (const imgElement of imageElements) {
  console.log(imgElement.src)
}
```



Being more specific

What if we wanted to select specific elements?

Give them all an ID!



```
<p id="special">Special</p>  
<p>Not special</p>  
<p id="special">Special</p>
```

But IDs are unique...
This wouldn't work



Solution: Classes

Like an ID, but multiple elements can share it!

```
<p class="special">Special</p>  
<p>Not special</p>  
<p class="special">Special</p>
```

Any element can be assigned a class, or multiple classes:

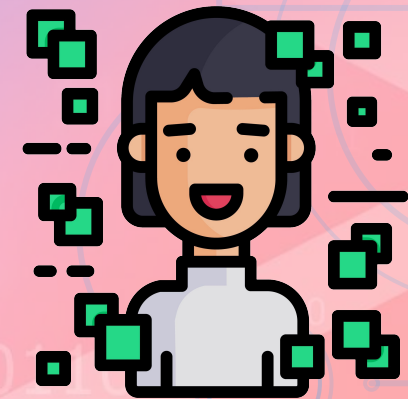
```
<a class="first-class second-class"></a>
```

Selecting Elements By Class

Then to select the special elements we would use:

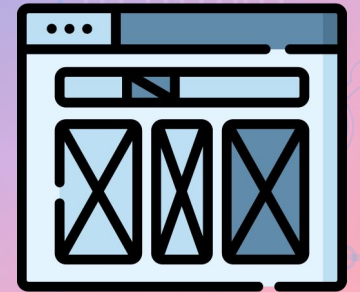
```
document.getElementsByClassName("special")
```

```
<p class="special">Special</p>  
<p>Not special</p>  
<p class="special">Special</p>
```



Chaining Selectors

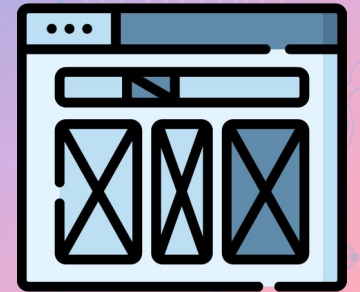
```
<div id="container">  
  <p class="special">Hello</p>  
  <p>world</p>  
</div>  
<p class="special"></p>
```



How would you select only the special p in the container?

Chaining Selectors

```
<div id="container">  
  <p class="special">Hello</p>  
  <p>world</p>  
</div>  
<p class="special"></p>
```



```
document.getElementById("container")  
  .getElementsByClassName("special")
```

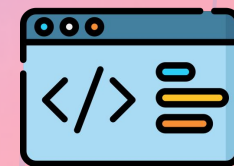
We control the DOM



Now we're kings and queens of the DOM!

We know how to select elements by:

- Their ID
- Their Class
- Their Tag

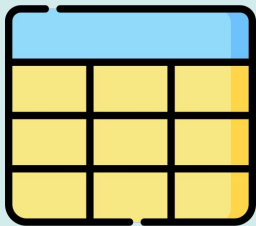


And how to select their child elements

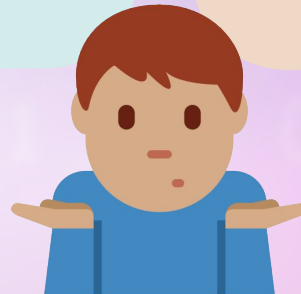
Back to Frequency Analysis

Now, how would you:

Populate a table with
the analysis result



Parse the HTML table
and use it as the key



Summary

Collection of <body> element children	<pre>document.body.children</pre>
Select an element using ID	<pre>document.getElementById("element-id")</pre>
Select first or last element and select using index	<pre>listEl.firstChild listEl.lastElementChild listEl.children[2]</pre>
Select element using tag name	<pre>document.getElementsByTagName("img")</pre>

Summary

Assign element an ID	<pre><p id="special">Special</p></pre>
Assign element a class	<pre><p class="special">Special</p></pre>
Select element using Class	<pre>document.getElementsByClassName("special")</pre>
Chain selection. Select using ID and Class	<pre>document.getElementById("container") .getElementsByClassName("special")</pre>

Questions?

Your Turn!

> Play around, have fun, ask questions!